

SYSTEM AND METHOD FOR A GLOBAL DIRECTORY SERVICE

Field of the Invention

The present invention relates to the provision of contact information between network subscribers. In particular, although not exclusively, the present invention relates to a system and method for a global directory service using electronic business card objects.

Discussion of the Background Art

10

The use of electronic business cards was originally proposed by the Versit Consortium in 1995 and subsequently taken up by the Internet Mail Consortium (IMC). The IMC has since developed a number of standards for electronic business cards such as the vCard or hCard, the latest of which is the open standard vCard v3.0. The vCard v3.0. is defined under two Internet Engineering Task Force (IETF) documents; i.e. under Request for Comments (RFC) 2425 (MIME Content-Type for Directory Information) and under RFC 2426 (MIME Directory Profile). As defined in the standards, both vCard and hCard can contain metadata, semantic information, graphics and images, and even audio or video clips. As defined in the standards, therefore, both vCard and the hCard are mere passive information elements.

20

One example of the use of vCards is discussed in European Patent Application No. EP 1589730 entitled "Method for Management of vCards". EP 1589730 is directed to a method for creating the vCards on a mobile terminal and exchanging the vCards between mobile terminal and other devices. Under the method of EP 1589730 the vCard is stored as image data in a JPEG file by inserting the vCard data in MIME type into the JPEG header.

25

US Patent No. 6442263 to *Beaton et al* entitled "Electronic Business Cards" discusses a method for providing electronic business cards for a communication device. Under the method of *Beaton*, business cards are created using CLID information, transferred among users of a telephone network, and used to initiate a call automatically.

30

Another system which utilises the vCard platform is disclosed in US Patent No. 7246099 to *Feldhahn*. *Feldhahn* allows individuals to maintain current contact information in another individual's contact software without having to individually
5 notify the receiving individual or manually resend the updated contact information to the individual. Static contact information for an individual is stored on a central server and is assigned a globally unique ID. The static contact information includes a dynamic link containing a creator's globally unique ID that may be utilized by recipients of the contact information to retrieve updated contact information.

10

US Patent No. 5493015 to *Desai* describes an electronic business card system which provides a compact and portable system to read and store business card data from business cards having computer readable data stored on computer readable storage media on the business cards. The electronic business card
15 system utilizes a reader coupled to a computer control system. The electronic business card system also provides organization and manipulation capabilities for the business card data accepted by the electronic business card system.

20

While electronic business card such as the vCard or hCard can be employed in a number of applications, the functionality of the electronic business cards is at present somewhat limited. It would be clearly advantageous to extend the functionality of such cards so that they include functions, scripts, and actionable elements. The present invention seeks to address such limitation of the existing electronic business card standards and implementation.

25

30

SUMMARY OF THE INVENTION

Disclosure of the Invention

Accordingly in one aspect of the present invention there is provided a method of
5 transferring contact information between a plurality of network subscribers, said method including the steps of:

compiling at a subscriber terminal the contact information into an electronic business card having one or more textual fields;

10 registering the electronic business card with a directory wherein the step of registering the electronic business card further includes:

encapsulating the electronic business card in an electronic business card object by mapping the textual fields of the electronic business card to one or more object attributes contained in the electronic business card object;

15 transmitting the electronic business card object to directory; and

storing the electronic business card object in a database

receiving at the directory a set of search parameters from a subscriber;

searching the database for electronic business card objects that match the search parameters

providing a listing of electronic business card objects to the subscriber;

20 receiving at the directory a request from the subscriber for an electronic business card object from the provided listing of electronic business card objects; and

forwarding the requested electronic business card object to the subscriber.

25 Suitably the classes of the electronic business card objects each include attributes and methods. A method is a function or routine that manipulates an attribute, utilises an attribute or performs some other function. The electronic business card objects may also include one or more actionable elements. These elements may include scripts that launch a native device application, launch an application or
30 service, and/or initiate a web-based or a wap-based interface. The electronic business card objects may include metadata, semantic information, graphics, images, and even audio or video clips.

In one embodiment of the present invention the electronic business card object may be a pointer to and a mask of a plurality of electronic business card objects, such that any user action on the electronic business card object acting as pointer to a plurality of electronic business card objects is effectively the same action individually on each of the electronic business card objects in such plurality of electronic business card objects.

Preferably the network is a mobile communications network, IP network or the like.

The step of registering may further include the step of validating a set of subscriber credentials. The subscriber credentials may be passive or active credentials or a combination of both. Passive credentials may include but not be limited to the mobile phone number of the requesting party and/or any such information which can be obtained and authenticated via SIM authentication from said party without its active knowledge or participation. Active credentials include but shall not be limited to a PIN, password, and/or any such information which a party has to explicitly provide.

Suitably the search parameters may be a set of keywords. In such instance the step of receiving may include parsing received keywords prior to initiating the appropriate database query using the said keywords.

The step of providing a listing of electronic business card objects may include the step of sorting and/or filtering the results of the query according to existing business rules. Such rules may include but are not limited to the relative weights accorded to the verified status of an object, confirmed relationships as attribute of an object, the level of keyword match precision, recency of information, Boolean operation, if any, and other considerations.

The method may further include the step of allowing a subscriber or any authorized party to update or delete any of the attributes of the electronic business card object. Suitably the step of updating the attributes of the electronic business card object may be performed via a plurality of user interfaces, including but is not limited to a

web or a wap interface, a local application, or a menu system using email, SMS, MMS, flash message, or an IP technology.

5 The method may also include the step of receiving a request for synchronization or download from a subscriber. Suitably the step of synchronizing further includes verifying the existence of the electronic business card object/s requested to be synchronized or downloaded and verifying the privacy or access level settings of the electronic business card object(s) requested to be synchronized or downloaded and sending an actionable notification to the owner of the electronic business card
10 object(s) requested to be synchronized or downloaded for the allowance or disallowance of the request, if the privacy or access level settings of electronic business card object(s) require consent prior to synchronization or download and sending a synchronization update or a copy of the electronic business card object(s) requested to be synchronized or downloaded to the requesting subscriber
15 on allowance of the synchronization or download request, or if the privacy or access level settings of such object(s) do not require consent prior to synchronization or download; and sending an appropriate notification to the requesting party.

20 The method may also include the step of verifying and authenticating the content of the electronic business card object. The step of verifying and authenticating may include tagging the object requested to be verified as certified on successful verification or authentication of the data in the electronic business card object and sending an appropriate notification to the concerned party / parties. Alternatively the step of verifying and authenticating may include automatically performing electronic
25 business card object verification by confidence level scoring with the use of an existing post-paid subscriber database, use of verified relationships associated with the object, use of profile information, and the like or a combination of any or all of these, subject to predefined business rules and subsequently tagging those objects yielding a confidence level equal to or higher than a predefined threshold, as
30 verified.

In a further aspect of the present invention there is provided a system for facilitating the transfer of contact information between network subscribers said system including:

at least one server coupled to the network;

5 at least one database coupled to the server;

a plurality of subscriber terminals coupled to the network wherein each subscriber's terminal is configured to send contact information associated with a subscriber to the server in response to a request by said subscriber; wherein the request causes the subscriber's terminal to compile the contact information into an
10 electronic business card having one or more textual fields and map the one or more textual fields of the electronic business card to one or more attributes of an electronic business card object and transmit the electronic business card object to the server for storage in the database.

15 Preferably the network is a mobile communications network, IP network or the like.

A subscriber may search and/or browse through the plurality of electronic business card objects stored in the database, in which the searching/browsing may be performed via a plurality of user interfaces, including but is not limited to a web or a
20 wap interface, a local application, or a menu system using email, SMS, MMS, flash message, or an IP technology.

The system may allow a subscriber or any authorized party to update or delete any of the attributes of the electronic business card object. Suitably the step of updating
25 the attributes of the electronic business card object may be performed via a plurality of user interfaces, including but is not limited to a web or a wap interface, a local application, or a menu system using email, SMS, MMS, flash message, or an IP technology.

30 The electronic business card objects stored in the database may be tagged as being a certified entry. A certified electronic business card object is one which the server provides a minimum level of certainty that the information in the business card object contains the information of the person or entity to which such business

card object purports to pertain. The certification of an electronic business card is based on a confidence level resulting from a verification process which could include, but is not limited to, requiring physical presence and submission of credentials, use of an existing post-paid subscriber database, use of verified
5 relationships associated with the object, or a combination of any or all of these, subject to certain business rules. Aside from the confidence level accorded to a verified electronic business card object, the verified status is likewise used to increase its relative weight in a search result, where a query has multiple exact and/or similar matches.

10

Suitably the system may be implemented as a distributed system across a plurality of networks. In such instance the global directory could include a plurality of computing platform coupled to each network within the plurality of networks, each computing platform being adapted to host, manage and maintain a global directory
15 service.

BRIEF DETAILS OF THE DRAWINGS

In order that this invention may be more readily understood and put into practical
20 effect, reference will now be made to the accompanying drawings, which illustrate preferred embodiments of the invention, and wherein:

FIG. 1 is a schematic diagram depicting a system for facilitating the transfer of contact information between network subscribers according to one embodiment of the present invention;

25 FIG. 2 depicting the process of updating an electronic business card object according to one embodiment of the present invention;

FIG. 3 is a schematic diagram depicting a search procedure according to one embodiment of the present invention;

30 FIG. 4 is a schematic diagram depicting the retrieval of an electronic business card object according to one embodiment of the present invention;

FIG. 5 is a schematic diagram depicting the process of verification and authentication an electronic business card object according to one embodiment of the present invention;

FIG. 6 is a schematic diagram depicting the process of synchronizing the electronic business card objects local on a mobile device with the retrieved card details according to one embodiment of the present invention;

5 FIG. 7A is a conceptual diagram of one possible architecture of an EBC object according to one embodiment of the present invention;

FIG. 7B is a schematic diagram depicting one possible menu structure produced from the EBC object architecture of FIG. 7A;

FIG. 7C is a conceptual diagram of one possible architecture of an EBC object according to one embodiment of the present invention;

10 FIG. 7D is a schematic diagram depicting one possible menu structure produced from the EBC object architecture of FIG. 7C;

FIG. 8A is a conceptual diagram of one possible architecture of an EBC object assigned a particular context according to one embodiment of the present invention;

15 FIG. 8B is a conceptual diagram of one possible architecture of an EBC object assigned a particular context according to one embodiment of the present invention;

20 FIG. 8C is a conceptual diagram of one possible architecture of an EBC object assigned a particular context according to one embodiment of the present invention;

FIG. 9 is a flow diagram representing a service discovery process employed in one embodiment of the present invention;

25 FIG. 10A is a conceptual diagram of one possible architecture of an EBC object assigned a particular context according to one embodiment of the present invention;

FIG. 10B is a conceptual diagram of the architecture of an EBC object assigned a particular context as viewed from a third party client device according to one aspect of the present invention;

30 FIG. 11 is a diagram representing a masking function according to one embodiment of the present invention; and

FIG. 12 is a flow diagram representing the implementation of a masking function according to a further embodiment of the present invention.

DESCRIPTION OF EMBODIMENTS OF THE INVENTION

With reference to Fig 1 there is illustrated a system for facilitating the transfer of contact information between a plurality of network subscribers **100** which the applicant has termed a global directory. As shown the global directory includes at least one server **101** coupled to a database **102** containing contact information for one or more registered parties **104**. As shown the server coupled to a host network **102** which provides a number of services to a plurality of subscribers **103₁**, **103₂**, ..., **103_n** including registration and retrieval of contact information from the global directory. Each subscriber **103₁**, **103₂**, ..., **103_n** can access the global directory to enter their contact information. Once registered, the contact details of the particular subscriber can be retrieved by any other subscriber **103₁**, **103₂**, ..., **103_n** via a query to the global directory service. The various registration and retrieval procedures etc are discussed in greater detail below.

While the above global directory system has been described in terms of a server coupled to a singular host network, it will be appreciated by one of skill in the art, that the global directory may also be implemented as a distributed system across a plurality of networks. In such instance, the global directory could include a plurality of computing platforms coupled to each network within the plurality of networks, each computing platform being adapted to host, manage and maintain a global directory service.

Fig 2 illustrates a registration procedure **200** that permits a subscriber to add or amend their contact details contained in the global directory according to one embodiment of the present invention. In order to register their details with the global directory, a subscriber is required to send their contact details (step **201**) in the form of an electronic business card (EBC) to the global directory service. Prior to transmission the EBC is encapsulated as an object (step **202**), hereinafter referred to as an EBC object. It is the EBC object that is then forwarded (step **203**) to the global directory service for registration. The encapsulation of the electronic business card into the electronic business card object is essentially a mapping of the various fields of the vCard, which are mere textual information, to textual

information attributes within the relevant attributes of the EBC object. At this point, the mapping utilises a one-to-one correspondence between vCard information and an EBC object attribute, or possibly a correspondence that is defined via business rule definition.

5

On receipt of the EBC object (step **204**) the global directory service proceeds to check if the EBC object is stored in its database (step **205**). If a record of the EBC object is found on the global directory it then proceeds to verify the credentials of the subscriber (step **206**). The credentials may include one or more of the following:

10 mobile phone number or SIM identity or IMEI of the requesting party and/or any such information which can be obtained from said party without its active knowledge or participation, a PIN, password and/or any such information which a party has to explicitly provide. If the credentials are sufficient, the user's previous entry within global directory's database is updated with the data contained in the

15 EBC object (step **207**). A notification of the successful update is sent (step **208**) to the user.

In the event there is no record of the EBC object within the database (i.e. a new subscriber requesting registration) the global directory service captures the user's

20 credentials (step **209**). Once the global directory service has extracted the relevant credentials the global directory then stores the EBC object in its database. Once the EBC object is stored the global directory it then proceeds to send a notification to the subscriber (step **208**). Such a new entry would be tagged as unvalidated or unverified by default.

25

Fig 3 depicts a process **300** for enabling a subscriber to search the information contained in the global directory according to one embodiment of the present invention. As shown, the search process is initialised by a subscriber entering a set of search parameters e.g. a set of keywords (step **301**). The search parameters are

30 then forwarded (step **302**) to the global directory. On receipt of the search parameters the global directory service proceeds to query its database (step **303**) for EBC objects containing attributes that match (step **304**) the search parameters.

In the event that there are EBC objects within the database that match the search parameters the directory service then compiles a sorted list of these matching EBC objects (step **305**) subject to privacy parameters. The sorted list is then forwarded (step **306**) to the subscriber for display on their terminal (step **308**). If there are no matching results based on step **304** from the database query **303** the global directory proceeds to send a notification to the subscriber indicating that no results were found (step **307**). The null result is then displayed to the subscriber's terminal **308**.

- 10 On receipt of the sorted listing of results based on step **306**, the subscriber can then retrieve the desired electronic business card object from the global directory (step **308**) subject to privacy parameters.

- 15 The privacy parameters which may be supported by the system provide for specific privacy settings which define the extent to which information from one's EBC object is made publicly available. Different levels of availability of information can be set and limited to different groups, each group defined by established relationships with the EBC object owner, degrees of relationship, common profile elements, and such other suitable distinction(s) that could be used as basis in calling a group a group.
- 20 Thus the EBC object forwarded to a subscriber may vary from subscriber to subscriber, depending on the different levels of privacy control applied for example a subscriber may belong to a trusted group of the EBC object owner, hence the EBC object that will be forwarded to such subscriber will contain information attributes and methods which may not be available to another subscriber from
- 25 another group.

- 30 Given the current trends in information technology, a person may have different phone numbers, different email addresses or similar information. This plurality of information may form the basis of different EBC objects configurations given a specific user's privacy settings or preferences, but all such information pertains to a single person. In such instances the GD may assign an internal Customer Reference Number (CRN) for the given EBC object owner. The CRN is then utilised to associate on-the-fly a particular EBC objects for the subscriber with the

relevant information. Thus, the phone number of a person may be different from the perspective of two subscribers having requested the same EBC objects, but both numbers will connect a phone call to the same person.

5 Since the CRN as discussed above uniquely identifies an EBC object owner, the system likewise makes possible the use of fun numbers. Fun numbers are numbers associated with both the EBC object owner and the subscriber requesting the EBC object. The fun number is generated on-the-fly, but is valid as a unique communications identifier (a phone number, an email address, a URL and the like)
10 between the EBC object owner and the requesting subscriber unless otherwise disabled by the EBC object owner. A different fun number is associated with each requesting subscriber, but this plurality of fun numbers is associated with a single EBC object owner. The fun number employs end-to-end association, hence the fun number is associated with both the CRN of the EBC object owner and the CRN of
15 the requesting subscriber. This allows for bidirectional communication between the two parties. Use of the fun number generated for a subscriber by a third party to connect to the EBC owner or subscriber associated with the fun number is not allowed as this Fun number system employs a built-in authentication mechanism. That is, only the person for whom a fun number has been generated can use the
20 fun number to connect to the person at the other end to whom the fun number is associated.

One example of the retrieval for process **400** for EBC objects from the GD according to one embodiment of the present invention is illustrated in Fig 4. As
25 shown the subscriber is presented with the sorted list of EBC objects from step **308** of Fig 3. The subscriber then selects the entry (step **401**) in the list for which they wish to retrieve the electronic business card object. The selection is then communicated to the global directory which then proceeds to look up the privacy settings (step **402**) assigned to the EBC object. In this particular case the EBC
30 objects are divided into two groups, i.e. private and public listings.

If the requested EBC object is determined to be a private listing based on step **403** the directory service then sends a request **404**, in the form of an actionable

message, to the owner of the EBC object (step **405**) for permission to disclose the information contained in the EBC object to the requesting party. The owner on receipt of the request then sends a response back to the directory service (step **406**). The directory service then determines whether the owner of the EBC object
5 has given permission to provide the requested information or denied the request (step **407**). If the owner has denied permission to provide the requested EBC object the global directory proceeds to send a denial notice (step **408**) to the requesting party **409**.

10 In the event that the owner of the EBC object has given permission to provide the requested information to the requesting party the directory service proceeds to retrieve the EBC object from its database (step **410**). Prior to sending the EBC object (step **412**) to the requesting party **409** the directory service may tag the EBC object (step **411**) as part of the back-up contact list for the requesting party. As can
15 be seen from Fig 4 these steps are also followed in the event that the EBC object is determined to be a public listing.

With reference to Fig 5 there is illustrated one possible configuration of a verification and authentication procedure **500** of electronic business card objects
20 stored within the global directory according to one embodiment of the present invention. As shown the global directory is configured to send a reminder to each subscriber registered with the directory i.e. each subscriber having a stored EBC object within the directory. On receipt of the reminder (step **501**) the subscriber has the option to proceed with authentication and verification or to opt out based on
25 step **502**. If the user opts out of the verification procedure the process is terminated.

In the event the user selects to proceed with the verification process the subscriber is presented with the option for manual or automatic verification (step **503**). If the
30 subscriber selects manual verification the user is required to physically attend a recognised Global Directory Wireless Data Centre **504** (e.g. mobile subscriber's retail data centre etc) and provide the suitable documentation. The wireless data centre then proceeds to send an authentication request for the user (step **505**) to

the global directory. By contrast, selecting the automatic authentication option causes an authentication request to be sent directly from the subscriber's mobile device (step **506**) to the global directory.

- 5 On receipt of an authentication request (step **507**) the global directory proceeds to check the credentials of the sending party (step **508**) to determine if the request was sent via a wireless data centre or user initiated (step **509**). If the request was sent by a Wireless Data Centre the EBC object is then tagged as being certified (step **510**) i.e. data contained in EBC has been independently verified and
10 authenticated. A notification of the successful verification and authentication is then sent to the subscriber (step **513**).

- If the global directory determines that the request was sent by the subscriber the global directory proceeds to check all available data (step **511**) regarding the EBC
15 object to verify and authenticate the identity of the requesting party. The data may include, but not limited to, established links and relationships between the EBC object with other certified EBC objects, information contained in the subscriber's Internet Service Provider (ISP) or Mobile Service Provider databases or other third party databases e.g. account information in a social network, utilities consumer
20 databases, etc. If the data obtained is sufficient based on decision step **512**, then the authentication request is granted and the EBC object is tagged as being certified (step **510**). A notification of the successful verification and authentication is then sent to the subscriber (step **513**). If the available data is not sufficient based on decision step **512** to permit the global directory to verify and authenticate the
25 identity of the requesting party a failure notification is then forwarded to the requesting party (step **513**).

- As can be seen from the above discussion the verification and authentication process is designed to provide a certain level of confidence that the information
30 contained in the EBC object is the correct information relating to a particular person or entity identified in the EBC object. Any action taken in relation to the information contained in a non-certified EBC object by a subscriber is at the subscriber's own

risk, as the global directory cannot provide any level of certainty that the information contained in non-certified EBC object is accurate.

One example of a process for synchronizing electronic business card objects stored
5 locally on a mobile device with the corresponding entries in the global directory according to one aspect of the present invention is shown in Fig 6. The synchronization process is initiated by subscriber sending a sync request (step **601**) to the global directory. On receipt of the sync request (step **602**) the global directory proceeds to retrieve all EBC objects stored on the subscriber's mobile device (step
10 **603**).

Once the global directory has retrieved the listing of EBC objects from the mobile device, the global directory begins to compare the entries in the retrieved listing of EBC objects with the EBC objects contained in its database. As shown, the global
15 directory compares the first entry in the retrieved listing of EBC objects with each EBC object stored in its database (step **604**). If there is no record of a given EBC object in the database (step **605**), the EBC object is considered as being a manually added entry and the global directory skips the entry (step **607**). The global directory then determines if there are more EBC objects within the listing of
20 retrieved EBC objects to be checked (step **611**) and if so the global directory proceeds to check the next EBC object in the listing of retrieved EBC objects (step **604**).

If at step **604** a matching record of the EBC object retrieved from the subscriber's
25 device is found, the global directory then compares the data contained in the retrieved EBC with the data contained in the EBC stored in its database (step **606**). If the data contained in each of the EBC objects is the same based on step **608**, then the information contained in the EBC on the subscriber's device is considered as being up-to-date and no further action is required and the entry is skipped (step
30 **609**). The global directory then determines if there are more EBC objects within the listing of retrieved EBC objects to be checked (step **611**) and if so the global directory proceeds to check the next EBC object in the listing of retrieved EBC objects (step **604**).

In the event that the data contained in the EBC object retrieved from the subscriber's device does not match the data contained in the EBC object contained in the global directory at step **608**, the directory considers the data in the EBC
 5 retrieved from the subscriber's device as being out of date and in need of updating. The global directory then sends an update to the subscriber containing the updated EBC object to the subscriber's device (step **610**). On receipt of the update (step **612**) the data contained in the EBC object stored on the subscriber's device is overwritten with the data contained in the notification. The global directory then
 10 determines if there are more EBC objects within the listing of retrieved EBC objects to be checked (step **611**) and if so the global directory proceeds to check the next EBC object in the listing of retrieved EBC objects (step **604**).

The synchronisation process as detailed above is repeated for each entry in the
 15 listing of EBC objects retrieved by the global directory until no more entries remain in the listing of EBC objects retrieved from the subscriber's device. At the conclusion of a successful full synchronization, the system may set a checkpoint object, so that subsequent synchronizations may be handled incrementally as opposed to performing a full scan.

20

As briefly mentioned above, the EBC object includes a number of attributes and methods, a conceptual diagram of the structure of one embodiment of an EBC object is depicted in Fig 7A. In this particular example, the EBC object includes two
 attributes **701₁**, **701₂** and two methods **702₁**, **702₂**. As mentioned above, the
 25 attributes are informational attributes and may be associated with various textual field of an entities virtual business card. Methods on other hand define actions which can be done to the EBC object.

In order to take advantage of the object orientated paradigm offered by the EBC
 30 object structure the applicant has developed a specialized menu interface. At present most menu systems at present are process oriented, i.e. menu systems are designed to reflect a sequence of user actions required to achieve a particular task.

The menu interface proposed by the applicant is an object centric menu system which focuses on what a user can do to a particular object.

An example of the menu system for interpreting the information stored in the EBC object in the context of a mobile handset is shown in Fig 7B. In this case the EBC object is read by the menu system's menu engine and displayed to the user as a phonebook entry **703**. In order to display the EBC object as a phonebook entry the menu engine determines the value assigned to the "name attribute" of the EBC object. As shown in the example of Fig 7A, the EBC object contains two attributes **701₁**, **701₂**. Either of these attributes may be assigned as the name attribute. In this example the first attribute **701₁** is the name attribute and contains the relevant information regarding the entity name e.g. "John Doe". The second attribute **701₂** in this instance is associated with the phone number of the entity. For ease of description, the attribute associated with an entities contact phone number or numbers will be hereinafter referred to as phone attribute.

As with a standard phone book entry item, the user is free to select the item to bring up a submenu **704**. The submenu **704** in this instance presents the user with a selection of the available methods associated with the relevant attributes of the EBC object. In this instance the methods **702₁** and **702₂** are assigned specific tasks "Call" and "Send Money". Method **702₁** in this instance is assigned the "Call" function and as such, the method is associated with attribute **701₂** i.e. the phone attribute. Selecting call on the menu display **704** causes method **702₁** to retrieve the relevant contact number from the phone attribute **701₂** and initiate a call to the relevant party. Thus no additional menu interaction is required on the part of the user to initiate the call. Similarly, selection of the "Send Money" option will cause method two **702₂** to initiate the relevant actions to effect transfer of an amount designated by the mobile user to an account specified in one of the attributes **701₁**, **701₂**.

As depicted the submenu **704** also includes a default menu item, "View Details" **707**. This menu item allows for viewing and editing of the values of the other

informational attributes, such as maybe an email address, birthday, and the like contained in one or more attributes of the EBC object.

While the examples depicted in Fig 7A and 7B utilize two attributes and methods, it will be appreciated by one of skill in art that the EBC object may include multiple attributes and methods. A conceptual diagram of the structure of an EBC object having multiple attributes and methods is shown in Fig 7C. As shown the Attributes can range from 1 to n , where n is an upper limit. Similarly, methods can be from 1 to m , where m is an upper limit.

Fig 7D depicts the menus generated by the menu engine for an EBC object having the structure depicted in Fig 7C. As with the examples discussed above, the menu engine in this instance translates the value contained in the name attribute of the EBC object and presents the information to the user in the form of a phone book entry **705**. Selection of the entry "John Doe" displays the submenu **706** containing a listing of the available methods **702₁, 702₂, ..., 702_m**. As shown the methods displayed as menu items are limited only to such number of methods minus one that could fit on screen. In the example above, while 5 methods as menu items will fit on screen, only 4 menu items corresponding method 1 to 4 (**702₁, 702₂, 702₃, 702₄**) are shown to give way to the required menu item "View Details" **707**. The menu engine then maps a key on the mobile device that enables scrolling to a next screen full of menu items. This could be use of the directional keys, a built-in joystick, or any hotkey designated for such purpose. Thus, the next screen full will show Method 5 to Method 9 with the last available screen displaying Method ($m - 4$) to Method m .

Under the applicant's proposed menu system, the EBC objects may also be provided with a non-visible attribute "Context". The context attribute defines a set of default attributes and methods associated with an EBC object. An example of the use of the context attribute is set out in Table 1 below. In this particular case 3 context types are provided, each context having a set of default attributes and methods.

Context	Attributes	Methods
Person	6	7
Institution	4	5
Device Owner	6	5

Table 1: Default Number of Attributes & Methods for Specific Contexts

It will be appreciated by those of skill in the art that the above table is merely illustrative and does not limit the definition of each context. It does not likewise limit the number of contexts which can be defined under the proposed menu system.

Conceptual diagrams of each of the EBC object structures for each of the above identified contexts are shown in Figs 8A, 8B and 8C. Fig 8A depicts one possible structure for the context classification "Person". Per table 1 above, the EBC object structure for this context includes six attributes **801** (including the non-visible context attribute **801₁**) and 7 methods **802**. The default attributes provide for this context include the name **801₂** and phone **801₃** attributes, as discussed above. In addition to these attributes, the EBC object structure also includes additional attributes which contain additional information concerning the person identified in the name attribute. These additional attributes can include, for example, an email attribute **801₄**, website attribute **801₅**, Blog attribute **801₆**. Each of the attributes **801** may be associated with one or more of the methods **802**. As shown the seven methods provided under the "Person" context may include such actions as messaging **802₁**, call **802₂**, visit site **802₃**, read blog **802₄**, poke **802₅**, send money **802₆** as well as the default view details **802₇**. The association of the various methods with the various attributes is discussed in greater detail below.

Fig 8B depicts one possible EBC object structure for the context classification "Institution". As shown, the EBC object structure for this particular context includes four attributes **801** and 5 methods **802**. The attributes **801** provided for this particular context include the name **801₂**, phone **801₃**, email **801₄** and the non-visible context attribute **801₁**. The methods **802** associated with the attributes **801** in this instance include in addition to the default view details method **802₇** and the

messaging **802₁** and call **802₂** as were provided under the “Person” context. In addition to these methods, the “Institution” context includes a number of methods specific to the Institution context, in this particular example these context specific methods are the “see catalog” **802₈** and “locate” **802₉** methods. As the name suggests, the “see catalog” **802₈** method is provided to allow the user to view the relevant product catalog for the institution identified in the associated name attribute, while the “Locate” **802₉** method provides information regarding the location of the relevant institution identified in the associated name attribute.

- One possible EBC object structure for the context “Device Owner” is illustrated in Fig 8C. As specified in table 1 above the Device Owner context includes six attributes **801** and five methods **802**. Thus, the focus in the Device Owner context is more on informational attributes **801** and less on methods **802**. As shown, the informational attributes **801** provided under the Device Owner context are similar to the informational attributes provided under the Person context. In this case the Device Owner context includes in addition to the non-visible context attribute **801₁**, the name **801₂**, phone **801₃**, email **801₄**, website **801₅**, and blog **801₆** attributes. The methods **802** provided under this particular context could include, for example, read notes **802₁₀**, manage money **802₁₁**, check planner **802₁₂** and write blog **802₁₃**. As in the case of the above discussed contexts, the Device Owner context also includes the default view details **802** method.

From the above examples, it can be seen that each specific EBC object context defines a default set of attributes and methods. It would therefore be easy to conclude that since a specific context has a set of default attributes and methods, the resulting menu system for EBC objects for each specific context would be the same. However, not all EBC objects of the same context will have all attribute fields fully populated.

- In order to determine which methods are to be displayed to the user for a given EBC object, the menu engine employs a service discovery process. In essence the menu engine compares a mapping of the attributes with the methods. An example of the service discovery process is depicted in Fig 9. As shown, the menu engine

firstly determines if the attribute mapped to the method exists (step **901**). If the attribute does not exist then the method is tagged as being unavailable (step **902**), conversely if the attribute exists then the method is tagged as being available (step **903**). The menu engine then determines whether there are any further methods to be checked (step **904**) for the particular EBC object and if so, the menu engine repeats process steps **901**, **902** or **903** (depending on availability of method) until there are no further methods to be checked at which point the menu engine generates a menu only containing menu (step **905**) items which corresponded to the methods tag as being available for the given EBC object.

As an example, consider EBC objects of the context "Person". Under this context the EBC object includes the "Read Blog" method which is mapped with the Blog attribute. The Blog attribute may have for its value either the Blog URL or null. It is this value that the menu engine reads during the step **901** to determine if the attribute exists. If the Blog attribute is null, the "Read Blog" method is not made part of the menu system associated EBC object. On the other hand, if there is a value corresponding to the Blog attribute, the "Read Blog" method becomes a menu item in the menu system associated with the particular EBC object.

While the above process of service discovery is performed on-the-fly as entries in the phonebook are manually created, the menu system resulting therefrom is not static. The menu engine may admit updates which can alter the appearance of the menu for a specific EBC object. Such updates may be triggered by any of the following events:

- A broadcast from the information owner
- Sharing of phonebook entry
- Synchronization with the global directory service (as described above)

A broadcast from the information owner happens when an information owner uses the mobile network to send out an updated copy of the EBC object that pertains to a particular information owner. An information owner is one locally defined on a mobile device as an EBC object with the context of a "Device Owner". Certain tools

are provided to such information owners to update and even change or alter the attributes and methods associated with the EBC object. The default set of attributes and methods associated with such EBC object is overridden by this customization. In the example illustrated in Fig 10A, an information owner (Person A) has edited the details of their EBC object such that it includes new attributes and methods. In this particular case Person A has added two new attributes "Social Net" **1001₁** and "Chat" **1001₂** to the default attribute fields of the non-visible context attribute **801₁**, the name **801₂**, phone **801₃**, email **801₄**, website **801₅**, and blog **801₆** attributes. Likewise, two new methods (see Social Net **1002₁**, and Chat **1002₂** have been added to the listing of default methods for the Device Owner context, namely read notes **802₁₀**, manage money **802₁₁**, check planner **802₁₂** and write blog **802₁₃**.

As will be appreciated by one of skill in the art that while the EBC object of person A is considered locally as belonging the "Device Owner" context, it is viewed by other people's device as belong to the context "Person". Thus person A's EBC object would be viewed locally by the device of person B as an EBC object having at least the default attributes and methods of the Person context discussed above. As far as person B is concerned, the EBC object pertaining to person A will remain without the attributes "Social Net" and "Chat" and consequently without the methods "See Social Net" and "Chat" until an update person A's EBC object occurs i.e. person B's menu engine admits update via receipt of broadcast from the information owner, Sharing of phonebook entry with person A, or person B request to sync their listing of EBC objects with the global directory service (as discussed in relation to Fig 6 above).

Sharing an EBC object is peer-to-peer, while broadcasting can be directed broadcast or simple broadcast. Directed broadcast is the simultaneous sending of his EBC object to a group of recipients, generally those selected from the phonebook of person A or any defined list of recipients. A simple broadcast, on the other hand, sends out the EBC to all possible recipients within a defined geographic region. In both cases, the effect is the same the recipient of the updated EBC object will have either the option to save the update or to reject it.

If the recipient opts to accept the updated EBC, then the menu engine will initiate a menu update on-the-fly, that is, when the updated EBC object as shown in Fig 10B is used. In the same example, if the recipient decides to access the updated EBC through his phonebook, the menu engine updates the menu system in the manner quite similar to the process illustrated through a service discovery process similar to that shown in Fig 9.

Aside from the menu update, the menu engine can also rearrange how the methods are presented as menu items. For each EBC object, the methods are ranked based on the following criteria:

- Success of transaction
- Frequency of transaction
- Cost of transaction
- Other business rules

In addition to the above, the methods which, based on the service discovery process described above, have been identified to be unavailable shall no longer be shown as part of the menu system, i.e. there is no menu item that corresponds to a method that invokes an unavailable service.

This adaptive rearrangement of menu items is done at all levels of a multi-level menu system. Thus, two EBC objects belonging to the same context will not have identically arranged set of menu items. Each EBC object will have a different arrangement of menu items based on how the device owner interacts with such object. A further criterion is included such that EBC objects which have not been used in user interactions for a predefined time period will have a default arrangement of menu items based on the general usage statistics of EBC objects of similarly context categories.

The menu system may also employ a masking function to allow users to interact with various individuals without the need to specifically identify each individual. The masking function makes use of a context classification known as the "Group"

context. The "Group" context is further divided into local or global classes. An EBC object of the context "local group" is one that serves as an alias to a group of EBC objects stored locally on the user's device. Any action on the alias is an action on all the EBC objects linked to the local group alias. This action, however, will be limited to those that can be done in a group, such as sending an SMS.

An example of the use of the masking function for the local group context is shown in Fig 11. As shown the SMS **1101** is addressed using the local group alias **1102**. The masking function proceeds to send the SMS to each individual identified in the EBC objects **1103₁**, **1103₂**, **1103₃**, **1103₄** contained in the local group.

An EBC object of the context "global group" may be similarly implemented. The difference, however, with a "local group" and a "global group" is that a "local group" is created by the device owner on the device, by masking EBC objects which are located on the device, while a "global group" is an EBC object from a global directory service which serves as a pointer to other EBC objects. The masked EBC objects in a "local group" are known to the device owner because instances of these masked objects are supposed to be stored in the device, while the masked EBC objects in a "global group" are not known to the device owner. The masked EBC objects in a "global group" are stored in the directory service database.

Fig 12 illustrates the use of the masking function for sending a message to an EBC objects in either the local or global group context. As shown the user sends the SMS to the group context alias (step **1201**). The masking function then proceeds to check if the group context attribute is a local group (step **1202**). If the group is determined to be a local group then the masking function proceeds to read the mask pointer attribute (step **1204**) which identifies all EBC objects contained in the group. The masking function then proceeds to extract the EBC objects from the local listing of EBC objects (step **1205**) and send the SMS to each of individual identified by the extracted EBC objects (step **1206**).

In the event that the group is a global group the masking function forwards the SMS to the global directory service (step **1203**). The global directory service then

proceeds to read the mask pointer (step **1204**) in order to extract the relevant EBC objects (step **1205**) from its database. Once the global directory has extracted the EBC objects contained in the group, it then proceeds to forward the SMS each of individual identified by the extracted EBC objects (step **1206**).

5

It is to be understood that the above embodiments have been provided only by way of exemplification of this invention, and that further modifications and improvements thereto, as would be apparent to persons skilled in the relevant art, are deemed to fall within the broad scope and ambit of the present invention described herein.

10

CLAIMS

1. A method of transferring contact information between a plurality of network subscribers said method including the steps of:

compiling at a subscriber terminal the contact information into an electronic
5 business card;

registering the electronic business card with a directory wherein the step of registering the electronic business card further includes:

encapsulating the electronic business card in an electronic business card object by mapping the textual fields of the electronic business card to one or
10 more attributes contained in the electronic business card object;

transmitting the electronic business card object to the directory; and

storing the electronic business card object in a database

receiving at the directory a set of search parameters from a subscriber;

searching the database for electronic business card objects that match the
15 search parameters

providing a listing of electronic business card objects to the subscriber;

receiving at the directory a request from the subscriber for an electronic business card object from the provided listing of electronic business card objects;
and

20 forwarding the requested electronic business card object to the subscriber.

2. The method of claim 1 wherein the step of registering further includes the step of validating a set of subscriber credentials.

25 3. The method of claim 2 wherein the subscriber credentials include the mobile phone number of the requesting party and/or any such information which can be obtained from said party without its active knowledge or participation.

30 4. The method of claim 2 wherein the subscriber credentials include at least one of the following: a PIN, password, and/or any such information which a party has to explicitly provide.

5. The method of any one of the proceeding claims wherein the search parameters are set of keywords.

6. The method of claim 5 wherein the step of receiving the search parameters further includes the step of parsing the received keywords prior to initiating the search.

7. The method of any one of the preceding claims wherein the step of providing a listing of electronic business card objects further includes the step of sorting and/or filtering the search results.

8. The method of claim 7 wherein sorting the listing of the electronic business card objects is in accordance with one of the following: a weighting accorded to the verified status of an object; confirmed relationships as attributes of an object; level of keyword match precision; recency of information or Boolean operation.

9. The method of any one of the preceding claims further including the step of allowing a subscriber or any authorized party to update or delete any of the attributes of the electronic business card object.

10. The method of claim 9 wherein the step of updating the attributes and/or methods of the electronic business card object is be performed via a plurality of user interfaces.

11. The method of claim 10 wherein the user interfaces includes at least one of the following: a web or a wap interface, a local application, or a menu system using email, SMS, MMS, flash message, or an IP technology.

12. The method of any one of the preceding claims further including the step of receiving a request for synchronization from a subscriber.

13. The method of claim 12 wherein the step of synchronization further includes the steps of:

verifying the existence of the electronic business card object/s requested to be synchronized

verifying the privacy or access level settings of the electronic business card object/s requested to be synchronized;

5 sending an actionable notification to the owner of the electronic business card object/s requested to be synchronized for the allowance or disallowance of the request based on the privacy or access level settings of electronic business card object/s;

 sending a synchronization update or a copy of the electronic business card
10 object/s requested to be synchronized or downloaded to the requesting subscriber on allowance of the synchronization request; and

 sending an appropriate notification to the requesting party.

14. The method of any one of the preceding claims further including the step of
15 verifying and authenticating the content of the electronic business card object.

15. The method of claim 14 wherein the step of verifying and authenticating further include the steps of:

 tagging the object requested to be verified as certified on successful
20 verification or authentication of the data in electronic business card object; and
 sending an appropriate notification to the concerned party / parties.

16. The method of claim 14 wherein the step of verifying and authenticating further include the steps of:

25 verifying the electronic business card object via confidence level scoring; and
 subsequently tagging those objects yielding a confidence level equal to or higher than a predefined threshold, as verified.

17. The method of claim 1 wherein the electronic business card object includes
30 one or more actionable element.

18. The method of claim 17 wherein the one or more actionable element includes scripts that launch a native device application, launch an application or service, and/or initiate a web-based or a wap-based interface,

5 19. The method of claim 1 wherein the electronic business card object is a pointer to and a mask of a plurality of electronic business card objects, such that any user action on the electronic business card object acting as pointer to a plurality of electronic business card objects is effectively the same action individually on each of the electronic business card objects in such plurality of electronic business
10 card objects.

20. A system for facilitating the transfer of contact information between network subscribers said system including:

at least one server coupled to the network;

15 at least one database coupled to the server;

a plurality of subscriber terminals coupled to the network wherein each subscriber's terminal is configured to send contact information associated with a subscriber to the server in response to a request by said subscriber; wherein the request causes the subscriber's terminal to compile the contact information into an
20 electronic business card object having one or more textual fields and map the one or more textual fields of the electronic business card to one or more object attributes contained in an electronic business card object and transmit the electronic business card object to the server for storage in the database.

25 21. The system of claim 20 wherein each subscriber terminal includes a menu engine adapted to read each object attribute contained in the electronic business card object and display a plurality of menus containing menu items representing one or more functions which can be performed in relation to the contact information represented by the electronic business card object.

30

22. The system of claim 21 wherein the electronic business card object is assigned a context and the menu engine is adapted to generate a set of default menu items for the assigned context.

23. The system of claim 22 wherein the menu engine is adapted to modify the set of default menu items for the context in response to modification of one or more object attributes of the electronic business card object by the electronic business card object owner.

24. The system of claim 20 wherein the electronic business card object includes one or more actionable element.

25. The system of claim 24 wherein the one or more actionable element includes scripts that launch a native device application, launch an application or service, and/or initiate a web-based or a wap-based interface,

26. The system of claim 20 wherein the electronic business card object is a pointer to and a mask of a plurality of electronic business card objects, such that any user action on the electronic business card object acting as pointer to a plurality of electronic business card objects is effectively the same action individually on each of the electronic business card objects in such plurality of electronic business card objects.